IMAGE BIT-DEPTH EXPANSION METHOD BASED ON SPARSE REPRESENTATIONS

Pavel Záviška
Doctoral Degree Programme (1), FEEC BUT
E-mail: xzavis01@vutbr.cz

Supervised by: Pavel Rajmic
E-mail: rajmic@vutbr.cz

Abstract: In this paper, a method for restoration of low bit-depth images based on sparse representations is presented. Proposed method is independent on the transform used, however for the purposes of experiments, Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are used. The experiments show that our method enhances the visual quality of low bit-depth images and performs better, in terms of PSNR, than basic bit-depth expansion methods.

Keywords: Image processing, Bit-depth expansion, Quantization, False Contours, Sparse Representations, Douglas-Rachford Algorithm

1 INTRODUCTION

The term bit-depth denotes the number of bits used to represent the color value of the pixel, i.e. the number of quantization levels. Standard bit-depth is usually 8 bits per channel (bpc) which means 24 bits per pixel (bpp) for the color image. However, some low-end devices capture only 16 bpp images, where Red, Green and Blue are represented by 5, 6, and 5 bits, correspondingly. Such devices usually suffer from poor image quality due to visual artifacts caused by low bit-depth (LBD). The main problems of LBD images are a sparse histogram, causing displaying only a part of all colors possible, and the visually annoying false contours artifact (also called posterization) that affects mainly low-frequency components of the image and displays hard transitions instead of smooth gradients.

To achieve a sufficient visual quality, it is necessary to perform a bit-depth expansion that approximates the original (non-quantized) image as closely as possible, which in the case of LBD image restoration reduces the false contour artifact as well.

In the past, several methods have been introduced to increase the bit-depth of images. Some of the simplest methods like Zero padding [6], Multiplication by an ideal gain [6] or Bit-replication [8] perform only color mapping which does not increase the number of used colors and therefore do not help to improve the contouring problem. Other, more sophisticated methods, are based on image filtering [1, 4, 7], flooding [3, 9], probability [5] or optimization [10].

In this paper, a method exploiting the sparsity prior is introduced and compared with basic bit-depth expansion methods.

2 SPARSE REPRESENTATIONS

Methods based on sparse representations assume that a signal $\mathbf{y} \in \mathbb{R}^N$ can be approximated by linear combination of a few columns of a dictionary $\mathbf{D} \in \mathbb{C}^{N\times P}$, i.e. $\mathbf{y} \approx \mathbf{Dx}$, where $\mathbf{x} \in \mathbb{C}^P$ has only a few non-zero coefficients (is sparse). These methods basically solve a linear equation $\mathbf{Dx} = \mathbf{y}$ with the requirement that $\mathbf{x}$ is sparse. That leads to optimization task

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_0 \quad \text{s.t.} \quad \mathbf{Dx} = \mathbf{y}. \tag{1}$$

This task can be extended by adding additional constraints depending on a specific application (denoising, inpainting, etc.). Images can be also processed by vectorizing a pixel matrix. These methods have found wide use in image processing, signal processing, machine learning, and many more applications.

Finding the sparsest solution of (1) is NP-hard, nevertheless, there exist several algorithms that approximate the solution. The computation cost is redeemed with a slight inaccuracy. In this paper, we use proximal algorithms [2], which are iterative algorithms designed to find a minimum of a sum of functions by sequential evaluation of proximal operators associated to each minimized function.

3 PROBLEM FORMULATION

Assume $\mathbf{y}$ represents the original analog signal that has been uniformly quantized with the quantization step α. The quantized signal will be denoted $\mathbf{y}_\mathrm{q}$. From the definition of uniform quantization can be assumed, that the samples of $\mathbf{y}$ must have lied in the range from $-\alpha/2$ to $\alpha/2$ from the respective quantization level. Thus, it is desirable to find samples fulfilling the following condition:

$$\forall i : -\alpha/2 \leq [\mathbf{y}_\mathrm{q}]_i - [\mathbf{Dx}]_i \leq \alpha/2, \tag{2}$$

where $[\cdot]_i$ denotes i-th element of the vector.

In this paper, we surrogate the non-convex ℓ_0-norm with the closest convex norm, which is the ℓ_1-norm. To enforce condition (2), we use the ℓ_∞-norm selecting the largest value of the vector, i.e. if the largest difference is smaller than $\alpha/2$, all other samples must fulfill the condition as well. This leads to following convex optimization problem:

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 \quad \text{s.t.} \quad \|\mathbf{y}_\mathrm{q} - \mathbf{Dx}\|_\infty \leq \alpha/2. \tag{3}$$

4 SPARSE DEQUANTIZATION ALGORITHM

4.1 DOUGLAS-RACHFORD ALGORITHM

To solve the optimization problem (3) by a proximal algorithm, it is necessary to convert this problem to the unconstrained form, where both requested properties (sparsity and additional constraint (2)) are expressed as a sum of two functions. The unconstrained form of optimization task (3) is:

$$\arg\min_{\mathbf{x}} \|\mathbf{x}\|_1 + \iota_C(\mathbf{x}), \tag{4}$$

where we define C as the set of all feasible solutions satisfying (2), i.e.

$$C = \{\mathbf{x} \mid \|\mathbf{y}_\mathrm{q} - \mathbf{Dx}\|_\infty \leq \alpha/2\} \tag{5}$$

and $\iota_C(\mathbf{x})$ denotes the indicator function returning zero if $\mathbf{x} \in C$ and ∞ otherwise. If projection onto C is available, the optimization task (4) can be solved by the Douglas-Rachford algorithm [2] shown in Algorithm 1.

Algorithm 1: Douglas-Rachford algorithm solving (4)

Input: Set starting point $\mathbf{x}_0 \in \mathbb{R}^N$.
Set parameters $\lambda = 1, \gamma > 0$.
for $i = 0, 1, \ldots$ **do**
  $\tilde{\mathbf{x}}^{(n)} = \mathrm{proj}_C \mathbf{x}^{(n)}$
  $\mathbf{x}^{(n+1)} = \mathbf{x}^{(n)} + \lambda \left(\mathrm{soft}_\gamma(2\tilde{\mathbf{x}}^{(n)} - \mathbf{x}^{(n)}) - \mathbf{x}^{(n)}\right)$
return $\mathbf{x}^{(n+1)}$

4.2 PROXIMAL OPERATORS

The Douglas-Rachford algorithm in Algorithm 1 uses two proximal operators in each iteration. Firstly, it is the soft thresholding $\mathrm{soft}_\lambda(\mathbf{c})$, which is a proximal operator of ℓ_1-norm, enforcing the reconstructed signal to be sparse. This operator is defined as:

$$\mathrm{soft}_\gamma(\mathbf{c}) = \begin{cases} \mathrm{sgn}(\mathbf{c})\,(|\mathbf{c}|-\gamma) & \text{if} \quad |\mathbf{c}| > \gamma, \\ 0 & \text{if} \quad |\mathbf{c}| \leq \gamma \end{cases} \tag{6}$$

and is applied element-wise to each element of $\mathbf{c}$. Secondly, a projection onto C is evaluated in each iteration, which is a proximal operator of indicator function ι_C, enforcing the reconstructed signal to fulfill (2). To find the projection $\mathrm{proj}_C(\mathbf{x})$ means to solve the optimization task

$$\arg\min_{\mathbf{z}} \|\mathbf{x}-\mathbf{z}\|_2 \quad \text{s.t.} \quad \mathbf{z} \in C, \tag{7}$$

that can be computed using a quadratic programming, for instance. This approach is possible, however, the computational cost is quite expensive. However, in the particular case of orthonormal basis by exploiting the unitary property, the task (7) can be written as:

$$\hat{\mathbf{z}} = \arg\min_{\mathbf{z}} \|\mathbf{D}\mathbf{x}-\mathbf{D}\mathbf{z}\|_2 \quad \text{s.t.} \quad \mathbf{z} \in C. \tag{8}$$

If we substitute $\mathbf{Dz}$ with $\mathbf{u}$, i.e. $\mathbf{z} = \mathbf{D}^{-1}\mathbf{u}$, we obtain the equivalent problem:

$$\hat{\mathbf{z}} = \mathbf{D}^{-1} \cdot \arg\min_{\mathbf{u}} \|\mathbf{D}\mathbf{x}-\mathbf{u}\|_2 \quad \text{s.t.} \quad \mathbf{u} \in \mathbf{D}\cdot C, \tag{9}$$

which can be rewritten into form:

$$\mathrm{proj}_C(\mathbf{x}) = \mathbf{D}^{-1} \cdot \mathrm{proj}_{\mathbf{D}\cdot C}(\mathbf{D}\mathbf{x}), \tag{10}$$

where the symbolic term $\mathrm{proj}_{\mathbf{D}\cdot C}(\mathbf{D}\mathbf{x})$ is simple element-wise mapping in the domain of pixels.

5 EXPERIMENTS AND RESULTS

For the experiments, we selected 5 different greyscale images, 512×512 pixels in size and 8 bpc bit-depth that are standard testing images in image processing (Baboon, Boat, Lena, Peppers and Sunset). These images represent either images with high details such as Baboon and also images with very slow gradients, such as Sunset.

We created LBD images ($\mathbf{I}_L$) from original 8 bpc images ($\mathbf{I}_H$) according to

$$\mathbf{I}_L = \left\lfloor \frac{\mathbf{I}_H}{2^{(H-L)}} \right\rfloor, \tag{11}$$

where H and L denotes bit-depth of original and LBD image, respectively. These LBD images were restored by the proposed algorithm with the Discrete Cosine Transform (DCT) and the Discrete Wavelet Transform (DWT) with the wavelet Daubechies5 as the dictionary $\mathbf{D}$. We performed 2 sets of tests for bit-depth expansion: 2 bpc to 8 bpc and 4 bpc to 8 bpc. Results of 4 to 8 bpc bit-depth expansion by proposed algorithm for Lena and Sunset images are presented in Fig. 1 and Fig. 2, respectively.

The quality of image restoration was evaluated by Peak Signal to Noise Ratio (PSNR) which for 8 bpc images is given by:

$$\mathrm{PSNR} = 10 \cdot \log_{10}\left(\frac{255^2}{\mathrm{MSE}}\right), \tag{12}$$

where MSE is a Mean Squared Error which is defined for two greyscale images $\mathbf{I}$ and $\mathbf{K}$ with $m \times n$ pixels as:

$$\mathrm{MSE} = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \|\mathbf{I}(i,j)-\mathbf{K}(i,j)\|^2. \tag{13}$$

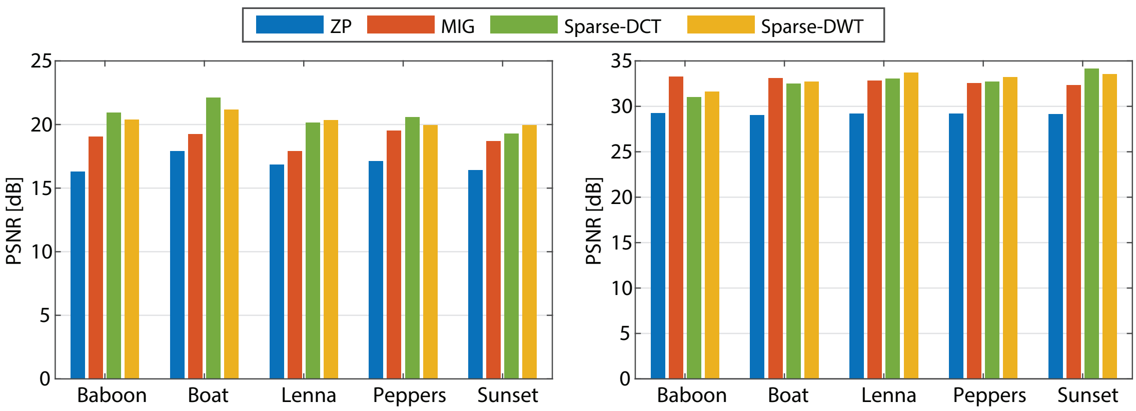
(a) Original 8-bit image **(b)** Quantized 4-bit image **(c)** Restored (DCT) **(d)** Restored (DWT)

Figure 1: Results of proposed method on Lena image.

(a) Original 8-bit image **(b)** Quantized 4-bit image **(c)** Restored (DCT) **(d)** Restored (DWT)

Figure 2: Results of proposed method on Sunset image.

Fig. 3 presents PSNR results for 5 tested images and bit-depth expansion from 2 to 8 bpc (3a) and from 4 to 8 bpc (3b). For the comparison purposes, the graphs also contain PSNR values of ZP and MIG methods. The graphs indicate that for highly distorted images (2 to 8 bits restoration), the DCT performs somewhat better than Wavelets. On the contrary, in the second test set (from 4 to 8 bpc) the DWT has obtained slightly higher PSNR values. Only when restoring high-detailed images (Baboon and Boat) from 4 bpc, MIG obtained the highest PSNR of all tested methods. The explanation of this effect is a fact that on high-detailed images the false contouring effect is not present, therefore a simple color mapping can be the most similar to the original image and thus get the highest PSNR values.



(a) 2 bpc $\rightarrow$ 8 bpc **(b)** 4 bpc $\rightarrow$ 8 bpc

Figure 3: PSNR results for bit-depth expansion for 5 testing images. Here ZP refers to Zero-Padding [6], MIG refers to Multiplication by an ideal gain [6], Sparse-DCT and Sparse-DWT refer to the proposed Sparse method with the usage of DCT and DWT basis as a dictionary.

6 CONCLUSION

In this work, a sparse method for restoring low bit-depth images, i.e. for bit-depth expansion was introduced. This method uses the Douglas-Rachford algorithm to solve the dequantization optimization problem. As a dictionary, practically any type of transform can be used, however, on account of the computational cost of the projection step, we used the DCT and the DWT orthonormal transforms. The proposed method can be used from any bit-depth to any bit-depth expansion. It can also be used for color images by applying on each component (R, G and B) separately.

Experiments performed on 5 greyscale images for bit-depth expansion from 2 and 4 to 8 bpc yield satisfactory results and indicate that the proposed method overcomes basic bit-depth expansion methods like ZP or MIG in terms of PSNR and also in subjective image quality.

Comparing two used transforms, the DCT slightly tends to add some noise to the restored images. On the contrary, the DWT smoothes some details from the restored image and sometimes adds some unwanted artifacts. Thus, further work might be devoted to finding an optimal dictionary for this purpose or to use redundant systems (frames) instead of the orthonormal basis.

REFERENCES

[1] AU, O. C.; FU, M. S.; WONG, P. H. W.; WONG J. W. C.; GUO, Z. Hybrid Inverse Halftoning using Adaptive Filtering. *Circuits and Systems* [online]. 1999, pp. 259–262, print ISBN: 0-7803-5471-0.

[2] COMBETTES, L. P.; PESQUET, J. C.: *Proximal splitting methods in signal processing*. Fixed-Point Algorithms for Inverse Problems in Science and Engineering, 2011, p. 28.

[3] CHENG, Ch.; AU, O. C.; LIU, Ch.; YIP, K. Bit-depth Expansion By Contour Region Reconstruction. *Circuits and Systems* [online]. 2009, pp. 944–947, E-ISBN: 978-1-4244-3828-0.

[4] LIU, Ch. H.; AU, O. C.; WONG P. H. W.; KUNG, M. C.; CHAO, S. Ch. Bit-Depth Expansion By Adaptive Filter. *Circuits and Systems* [online]. 2008, pp. 496–499, E-ISBN: 978-1-4244-1684-4.

[5] MITTAL, G.; JAKHETIYA, V.; JAISWAL, S. P.; AU, O. C.; TIWARI, A. K.; WEI, D. Bit-Depth Expansion Using Minimum Risk Based Classification. *Visual Communications and Image Processing (VCIP)* [online]. 2012, pp. 1–5, E-ISBN: 978-1-4673-4406-7.

[6] QIAO, M.; WANG, W.; NG, M. K. A Variational Method for Expanding the Bit-Depth of Low Contrast Image. *Energy minimization methods in computer vision and pattern recognition: 9th International Conference, EMMCVPR 2013* Lund, Sweden. 1st edition. 2013, pp. 54–65, ISBN: 3642403948.

[7] TAGUCHI, A.; NISHIYAMA, J. Bit-Length Expansion By Inverse Quantization Process. *Signal Processing Conference (EUSIPCO)* [online]. 2012, pp. 1543–1547, ISSN: 2219-5491.

[8] ULICHNEY, R.; CHEUNG, S. Pixel Bit-Depth Increase by Bit Replication. *Proc. Society of Photo-Optical Instrumentation Engineers* [online]. 1998.

[9] WAN, P.; AU, O. C.; TANG, K.; GUO, Y.; FANG, L. From 2D Extrapolation to 1D Interpolation: Content Adaptive Bit-Depth Expansion. *2012 IEEE International Conference on Multimedia and Expo (ICME)* [online]. 2012, pp. 170–175, ISSN: 1945-7871.

[10] WAN, P.; CHEUNG, G.; FLORENCIO, D.; ZHANG, C.; AU, O. C. Image bit-depth enhancement via maximum-a-posteriori estimation of graph AC component. *2014 IEEE International Conference on Image Processing (ICIP)* [online]. Paris, 2014, pp. 4052–4056.